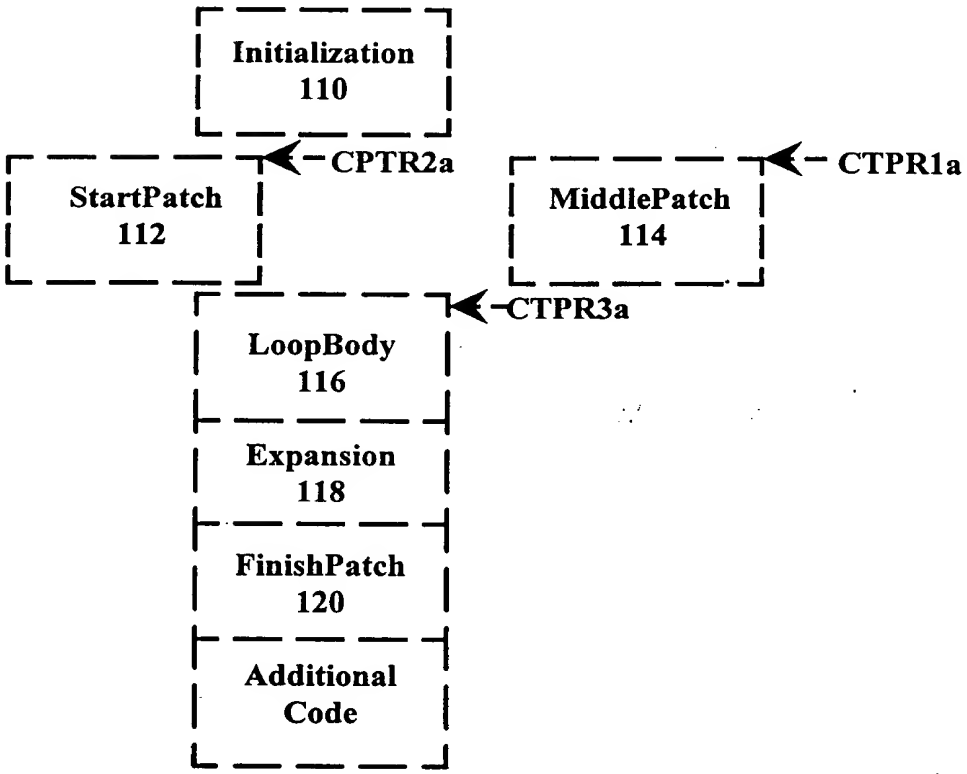

Initialization
110
CPTR2a
CTPR1a
StartPatch
112
MiddlePatch
114
CTPR3a
LoopBody
116
Expansion
118
FinishPatch
120
Additional
Code

FIG. 1

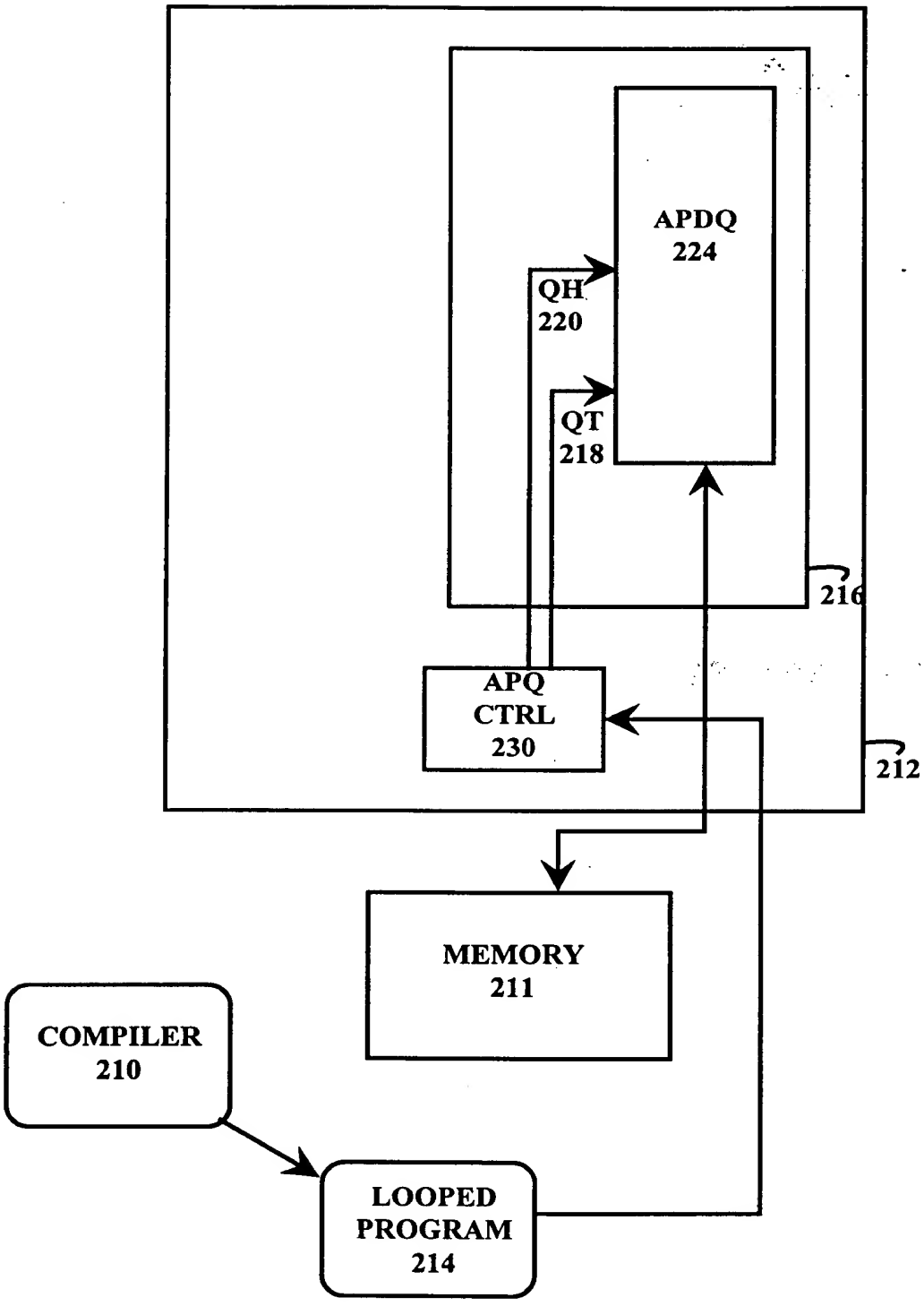

200
APDQ
224
QH
220
QT
218
216
APQ
CTRL
230
212
MEMORY
211
COMPILER
210
LOOPED
PROGRAM
214

FIG. 2

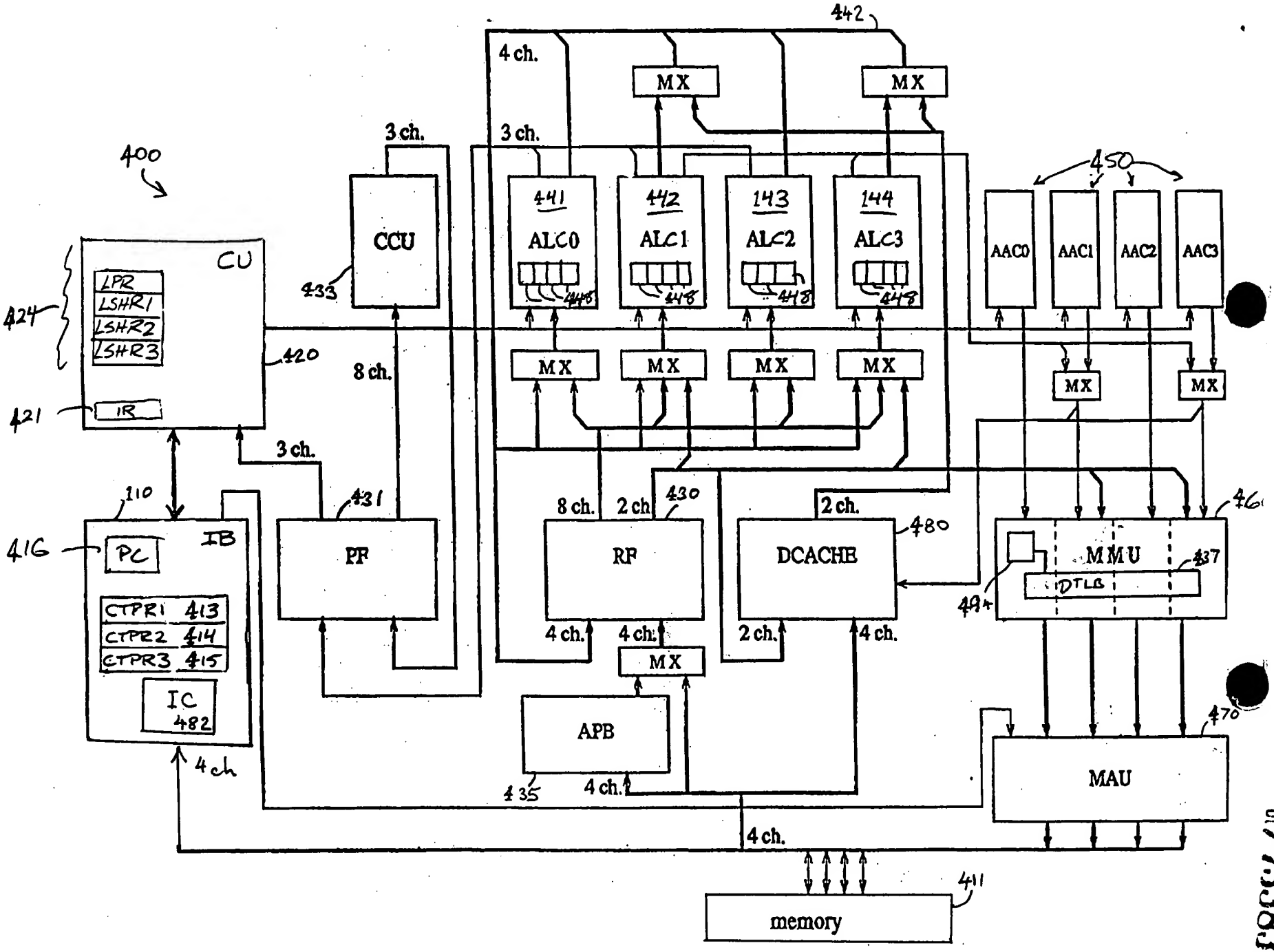
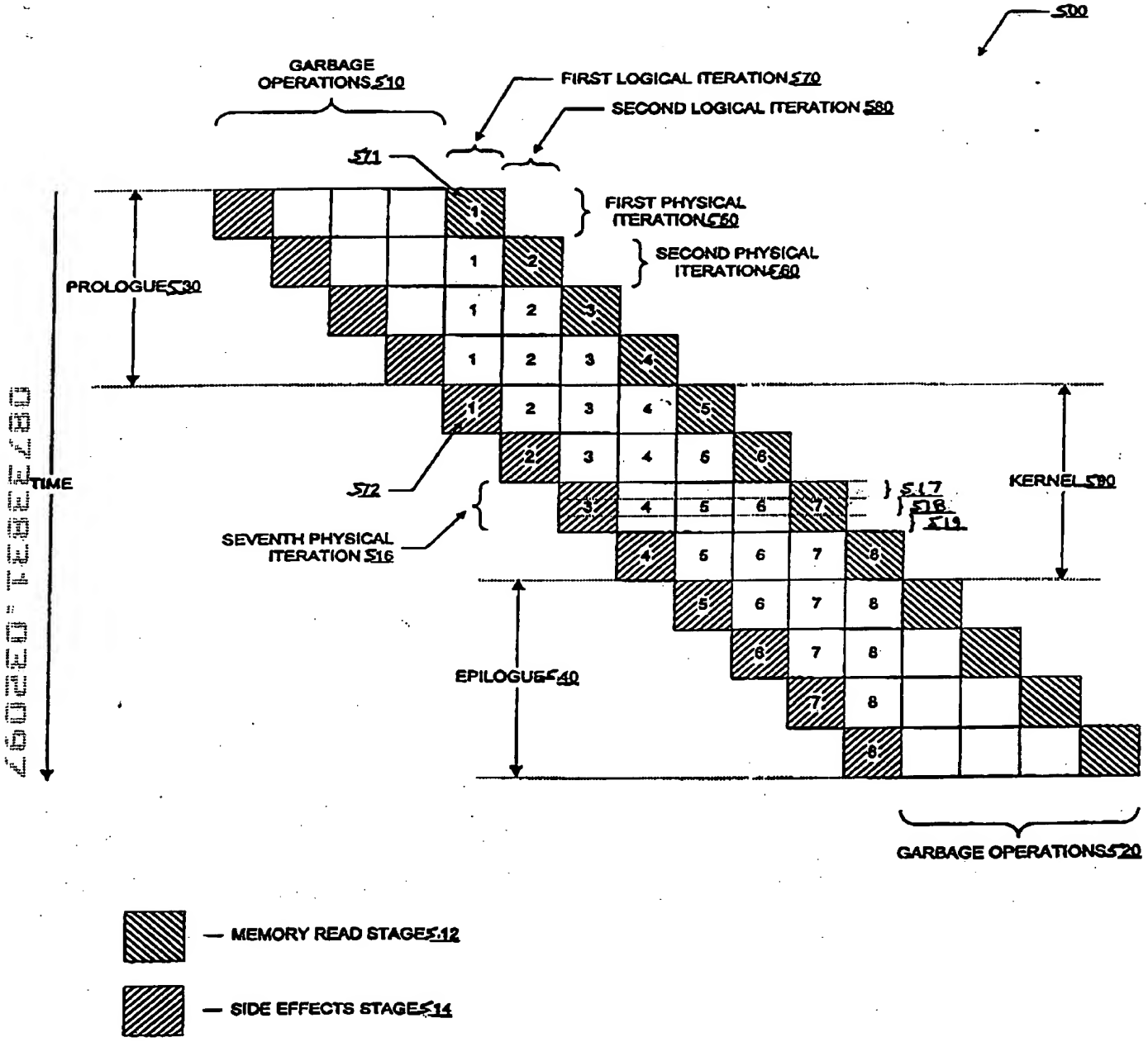


FIGURE 3

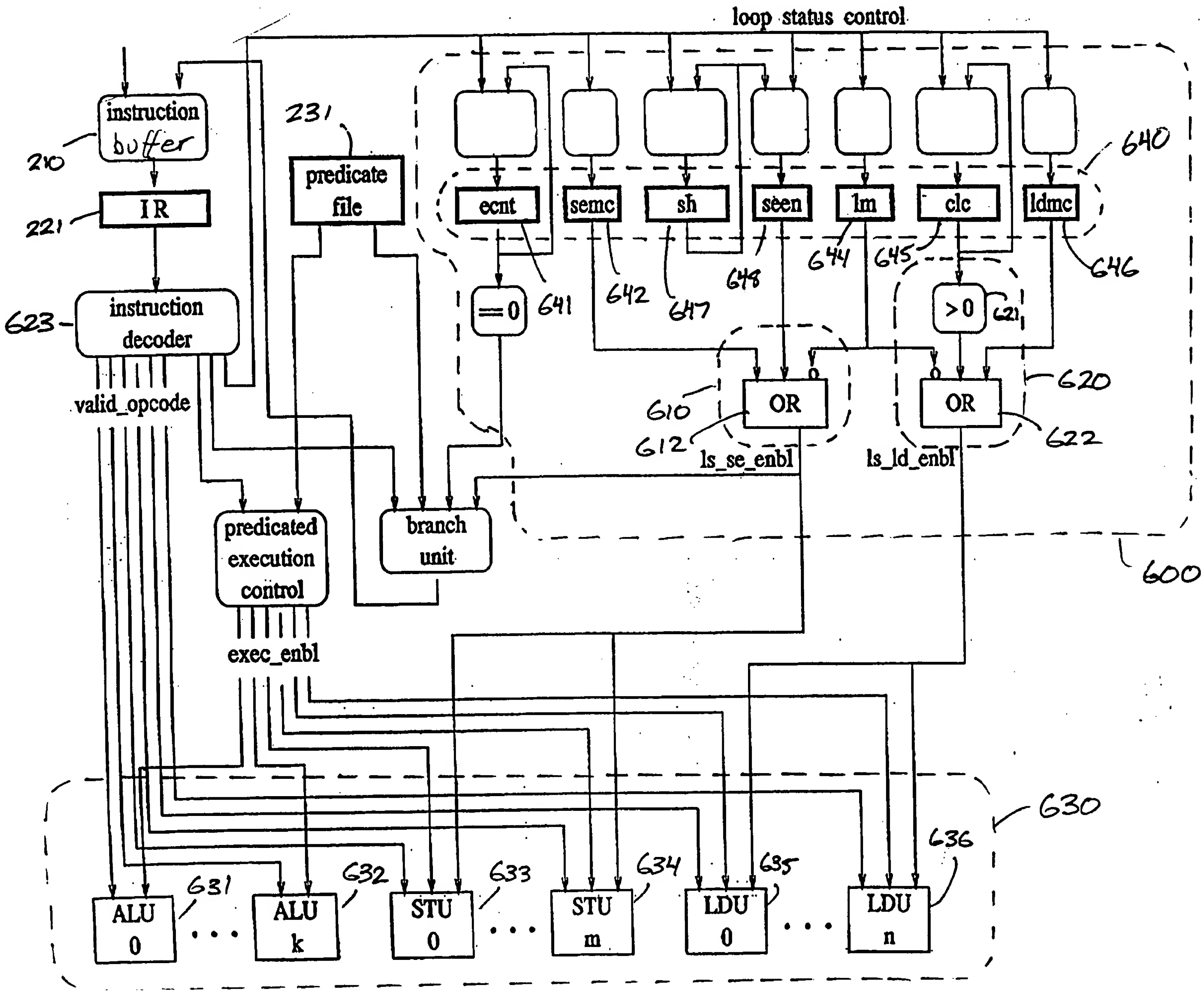


— MEMORY READ STAGE 512

— SIDE EFFECTS STAGE 514

FIGURE 4

FIGURE 5



08/733831 6

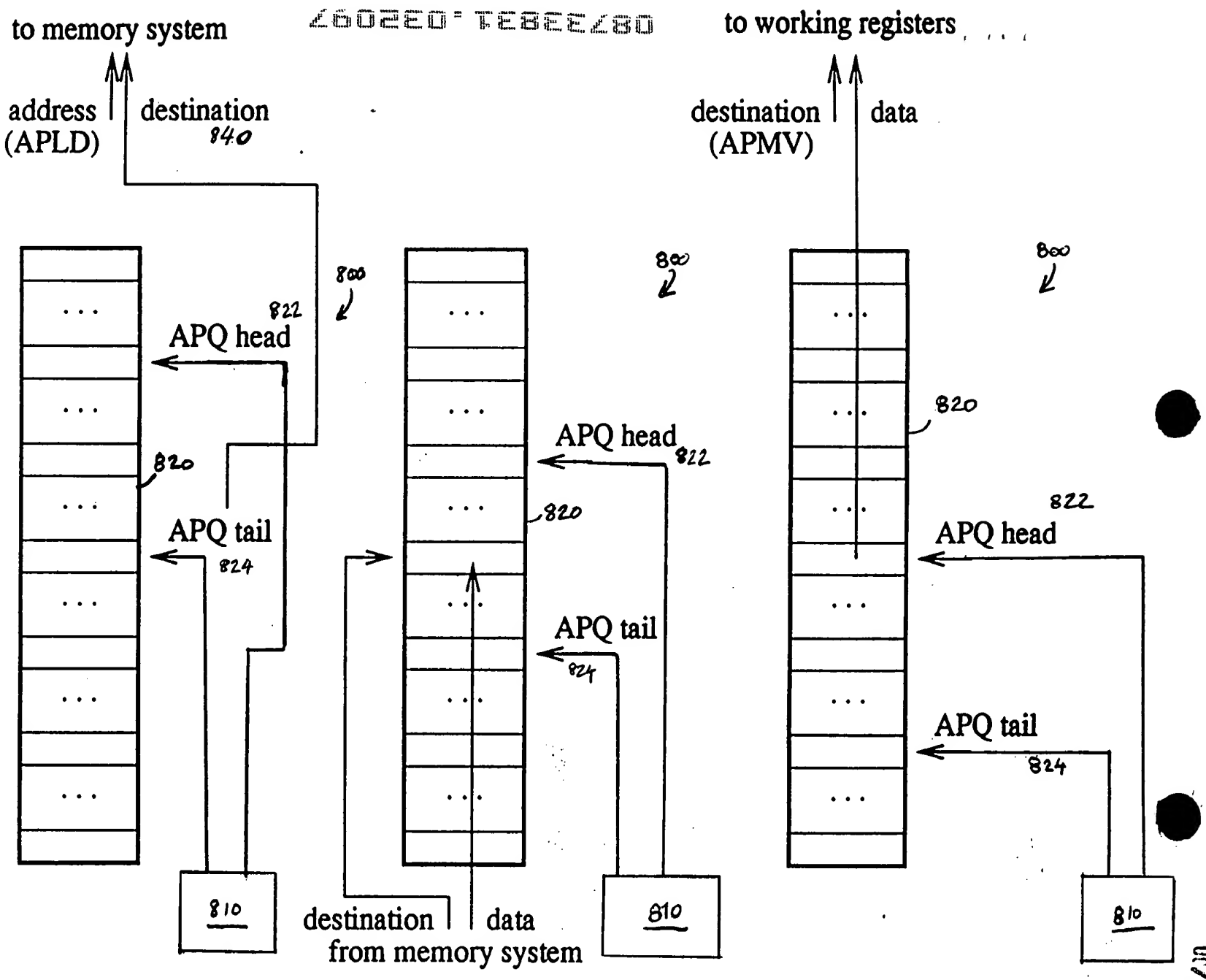


FIG. 6a)

FIG. 6b)

FIG. 6c)

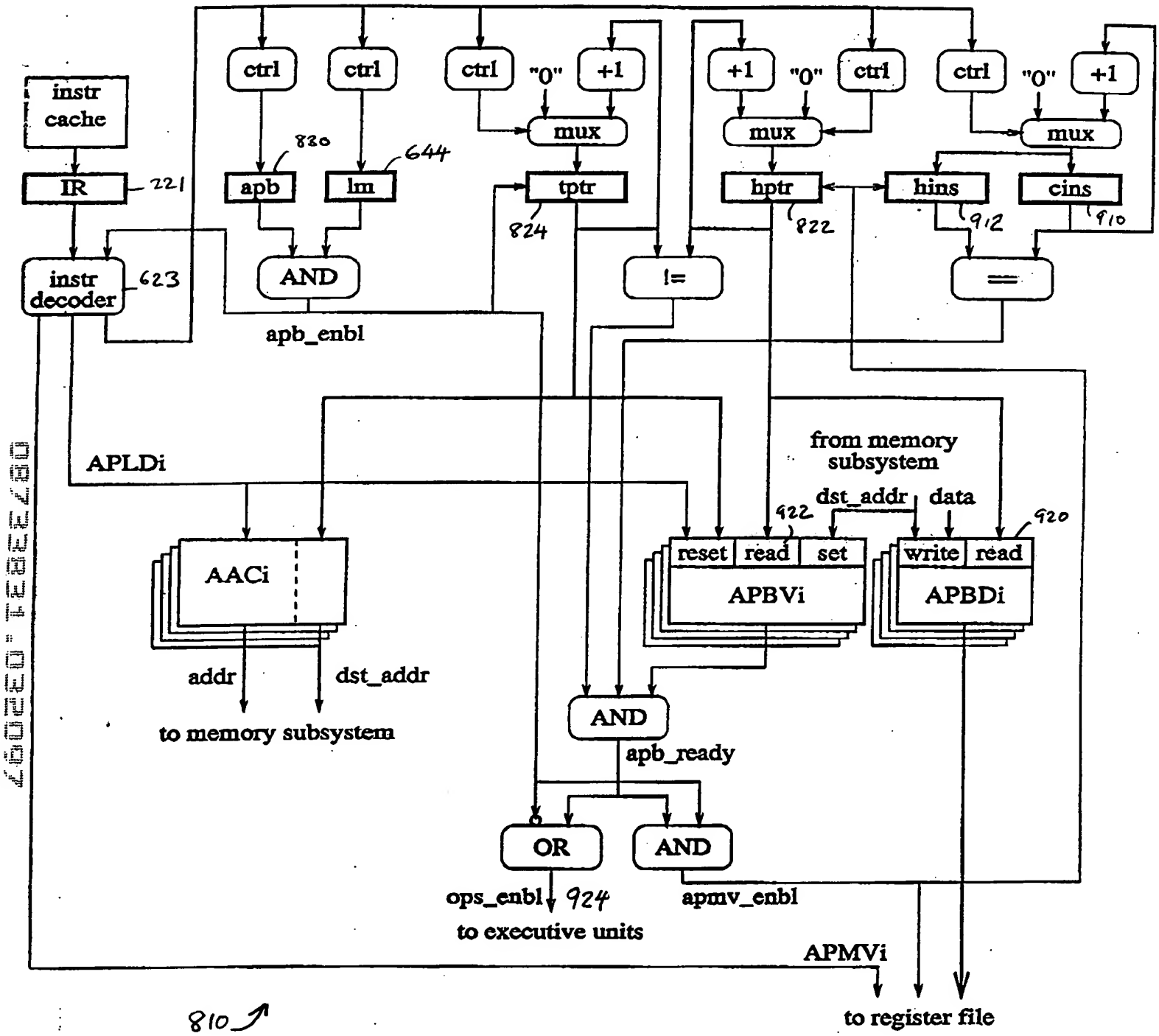

instr cache
IR
221
instr decoder
623
ctrl
ctrl
ctrl
"0"
+1
mux
tptr
824
apb
820
lm
644
AND
apb_enbl
!=
+1
"0"
ctrl
mux
hptr
822
ctrl
"0"
+1
mux
hins
912
cins
910
==
APLDi
from memory subsystem
dst_addr
data
922
920
AACi
reset
read
set
APBVi
write
read
APBDi
addr
dst_addr
to memory subsystem
AND
apb_ready
OR
AND
ops_enbl
924
to executive units
apmv_enbl
APMVi
to register file
810

fig. 7

PI	clc	ecnt	cldovl	tptr	hptr	cins	hins	apb_ready	logical iterations 0	1	2	3	memory data
				0	0	0	0	0	APLD a0 -> APB[0]				
1	4	1	0	1	0	1	0	0	APLD b0 -> APB[1]				
				2	0	2	0	0	-				
				3	0	3	0	0	-				
				4	0	0	0	0	-	APLD a1 -> APB[4]			
2	3	1	1	5	0	1	0	0	-	APLD b1 -> APB[5]			
				6	0	2	0	0	-	-			APB[0] = a0
				7	0	3	0	0	-	-			
				8	0	0	0	1	APMV a0	-	APLD a2 -> APB[8]		APB[1] = b0
3	2	1	2	9	1	1	1	1	APMV b0	-	APLD b2 -> APB[9]		
				10	2	2	2	1	FMUL	-	-		APB[4] = a1
				11	3	3	3	1	-	-	-		
				12	4	0	0	1	-	APMV a1	-	APLD a3 -> APB[12]	APB[5] = b1
4	1	1	2	13	5	1	1	1	-	APMV b1	-	APLD b3 -> APB[13]	
				14	6	2	2	1	-	FMUL	-	-	APB[8] = a2
				15	7	3	3	1	STORE c0	-	-	-	
				16	8	0	0	1		-	APMV a2	-	
5	0	1	2	17	9	1	1	0		-	-	-	
				18	9	2	1	0		-	-	-	APB[12] = a3
				19	9	3	1	0		-	-	-	APB[9] = b2
				20	9	0	1	0		-	-	-	APB[13] = b3
6	0	0	3	20	9	1	1	1		-	APMV b2	-	
				20	10	2	2	1		-	FMUL	-	
				20	11	3	3	1		STORE c1	-	-	
				20	12	0	0	1			-	APMV a3	
7	0	0	2	20	13	1	1	1			-	APMV b3	
				20	14	2	2	1			-	FMUL	
				20	15	3	3	1			STORE c2	-	
				20	16	0	0	1				-	
8	0	0	1	20	17	1	1	1				-	
				20	18	2	2	1				-	
				20	19	3	3	1				STORE c3	

FIG. 8